Saliency Detection via Absorbing Markov Chain With Learnt Transition Probability

Lihe Zhang, Jianwu Ai, Bowen Jiang, Huchuan Lu, *Senior Member, IEEE*, and Xiukui Li

***Abstract*— In this paper, we propose a bottom–up saliency model based on absorbing Markov chain (AMC). First, a sparsely connected graph is constructed to capture the local context information of each node. All image boundary nodes and other nodes are, respectively, treated as the absorbing nodes and transient nodes in the absorbing Markov chain. Then, the expected number of times from each transient node to all other transient nodes can be used to represent the saliency value of this node. The absorbed time depends on the weights on the path and their spatial coordinates, which are completely encoded in the transition probability matrix. Considering the importance of this matrix, we adopt different hierarchies of deep features extracted from fully convolutional networks and learn a transition probability matrix, which is called learnt transition probability matrix. Although the performance is significantly promoted, salient objects are not uniformly highlighted very well. To solve this problem, an angular embedding technique is investigated to refine the saliency results. Based on pairwise local orderings, which are produced by the saliency maps of AMC and boundary maps, we rearrange the global orderings (saliency value) of all nodes. Extensive experiments demonstrate that the proposed algorithm outperforms the state-of-the-art methods on six publicly available benchmark data sets.**

***Index Terms*— Salient object detection, absorbing Markov chain, transition probability matrix, angular embedding.**

I. Introduction

SALIENCY detection simulates human visual attention, which involves two main research directions: fixation prediction and salient object detection. The former focuses on matching human eye movements and the gaze on the whole object is not necessarily consistent, whereas the latter aims at completely detecting salient objects, which expects that the whole object can be uniformly highlighted. In recent years, salient object detection has been applied to many computer vision tasks, such as image retrieval [1], image segmentation [2], image classification [3], object recognition [4], to name a few. Although much significant progress has been made, salient object detection remains a challenging problem.

Existing salient object detection methods are categorized as bottom-up (data-driven) models [5]–[11] and top-down (task-driven) models [12]–[16]. The former has a wide range of application because it mainly depends on some low-level visual features (*e.g.*, color, intensity or orientation) and some prior knowledge (*e.g.*, contrast, compactness, uniqueness or boundary).

On the contrary, top-down models, including CNN-based deep learning ones, capture representative high-level features, thereby detecting salient objects of certain sizes and categories. In general, their performance is better than that of bottom-up models. However, top-down methods often need the time-consuming training process.

In our previous work [17], we propose a bottom-up saliency estimation approach, which formulates salient object detection as a random walk problem in the absorbing Markov chain (AMC). Given an image graph as a Markov chain and some absorbing nodes (background nodes), the absorbed time starting from object nodes is longer than that from background nodes. Therefore, the length of the absorbed time can be computed as saliency value. However, this time is not always effective especially when there are long-range (i.e. large area) smooth background regions near the image center, as the nodes in this kind of regions have small probabilities of randomly walking to absorbing nodes. To this end, the equilibrium probability is used to regulate the absorbed time in [17]. This method heuristically refines saliency maps by utilizing equilibrium probability when the maps present long-range regions with mid-level saliency (i.e. saliency value close to 0.5). Therefore, it difficultly achieves the optimized results for all images, thereby reducing the overall robustness of saliency detection.

In this work, we deeply consider the importance of transition probability matrix in computing absorbed time. When we use boundary nodes as absorbing nodes, the random walk starting from background nodes can easily reach the absorbing nodes, because the transition probabilities between them are large. While it is difficult to reach absorbing nodes for a random walk starting from object nodes because their transition probabilities are small and their spatial distances are long. Inspired by this observation, we learn to compute the transition probability matrix to improve the saliency performance.

To obtain the learnt transition probability matrix, motivated by [18], we employ the *sparse-to-full* based method. The *sparse* means that we first construct a sparsely connected graph, which primely considers the local context information

Manuscript received April 12, 2017; revised August 2, 2017 and October 18, 2017; accepted October 21, 2017. Date of publication October 26, 2017; date of current version November 28, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61371157, Grant 61472060, Grant 61528101, and Grant 61725202, and in part by the Fundamental Research Funds for the Central Universities under Grant DUT2017TB04 and Grant DUT17TD03. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shuicheng Yan. *(Corresponding author: Lihe Zhang.)*

L. Zhang, J. Ai, H. Lu, and X. Li are with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: zhanglihe@dlut.edu.cn; aijianwu@mail.dlut.edu.cn; lhchuan@dlut.edu.cn; xli@dlut.edu.cn).

B. Jiang is with AutoNavi Software Company, Ltd., Beijing 100102, China (e-mail: pengwen314@163.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2017.2766787

of each node. The *full* means that a dense affinity matrix is learnt based on the sparse graph, which can increase the transition probabilities between absorbing nodes and all transient nodes, thereby reducing the absorbed time of transient nodes (especially the nodes in long-range smooth background regions near the image center). In addition, we adopt different hierarchies of deep features extracted from FCN-32S [19], which describe both low-level and high-level image cues. Considering the importance of different kinds of features, we utilize every kind of feature to compute a sparse affinity matrix and fuse them to infer a full learnt transition probability matrix in a unified framework.

The full transition probability matrix also reduces the absorbed time of salient nodes, which possibly decreases the saliency of partial salient regions. To address this problem, a novel refinement model based on angular embedding (AE) algorithm is proposed in this work. The goal of AE is to find the global orderings based on pairwise local orderings. To precisely determine the local orderings between foreground and background, we propose two criterions based on saliency maps produced by AMC and boundary maps produced by [20]. Given the local orderings between all pairs of nodes, their saliency values are rearranged and refined. The experimental results show that the refinement model is very effective.

The main contributions of this work are listed as follows:

(1) We propose an absorbing Markov chain model for saliency detection, which achieves a learnt transition probability matrix by the *sparse-to-full* method combined with multiple-layer deep features.

(2) To the best of our knowledge, angular embedding technique is first imported into saliency detection, which can promote the perceptually homogeneous regions to get the same saliency and enhance the saliency of the whole salient object.

(3) It is demonstrated that the proposed algorithm performs favorably against the state-of-the-art saliency detection methods on six benchmark datasets.

The remainder of this paper is organized as follows: Section II discusses some previous works related to this paper. In Section III, the absorbing Markov model with learnt transition probability matrix is proposed. The principle of angular embedding and its refinement application in saliency detection are detailed in Section IV. Section V demonstrates and analyzes the experimental results. Finally, the whole paper is concluded in Section VI.

II. Related Work

The proposed algorithm is an absorbing Markov based bottom-up model, which uses fully convolutional network features at the same time. Therefore, this section is introduced in two aspects, including traditional models, in which some previous Markov related methods are detailed, and the CNN-based deep learning models.

To date, numerous traditional saliency methods have been proposed and obtained good detection performances. Different methods characterize this problem varying from one perspective to another, such as contrast [5], [6], [21], spatial compactness [7], [22], objectness [5], [23], edge density [24], distinct patterns [25] and geometry information [9]. Some methods simulate saliency detection in Bayesian inference models [26], [27] and sparse reconstruction [28]–[30]. Some works based on background prior [31], [32] aim to compute a distance map with respect to a set of background seed nodes. Long and Liu [33] evaluate the qualities of many results produced by different saliency methods to get the best saliency map.

Some diffusion-based models detect salient objects by selecting seeds and propagating saliency labeling, which are related to our work. Yang *et al.* [8], [34] compute saliency by ranking the similarities of all nodes with background and foreground seeds in a two-stage scheme. Wang *et al.* [35] improve the detection performance by exploiting a novel graph structure and removing wrong boundary seeds. Gong *et al.* [36] propose the teaching-to-learn and learning-to-teach strategies to improve the propagation quality. In addition, Jiang *et al.* [37] propose a general scheme to promote any diffusion-based saliency model by re-synthesizing the diffusion matrix.

There are also many methods based on Markov chain [38]–[43]. Costa [38] rely on the frequency of visits to each node at the equilibrium state to calculate a saliency map. Harel *et al.* [39] extend the above method by re-defining transition probability between two nodes. These approaches based on equilibrium distribution often only highlight boundary regions rather than the entire object. Afterwards, Gopalakrishnan *et al.* [40] exploit the hitting time to find the most salient seed on multiple graphs, which prefers to highlight the global rare regions and does not suppress image backgrounds well, thereby decreasing the overall saliency of objects. Kim *et al.* [41] develop a coarse-to-fine refinement model based on the random walk with restart, in which the stationary distribution at a coarse scale is used as the restarting distribution at a fine scale. Li *et al.* [42] formulate saliency detection as a regularized random walks ranking problem. They design a fitting constraint to restrict that the final saliency map should not differ too much from the prior saliency distribution. Besides, Sun *et al.* [43] identify the discriminative region by using the Markov absorption probability, which means the probability that a transient node is absorbed by an absorbing node. This method regards background and salient nodes as absorbing nodes in a two-stage scheme. There exists a well-known problem that the accurate selection of salient nodes is difficult, especially for complex scenes. Different from [43], in this work, the saliency of a node is represented as the expected number of times from this node to all other transient nodes before absorption. And we naturally use boundary nodes as absorbing nodes. The most important is that we obtain a learnt transition probability matrix by combining multiple features to improve the accuracy of random walk.

Numerous deep learning based methods spring out recently and achieve the state-of-the-art saliency detection performance. Wang *et al.* [44] and Li and Yu [45] use deep features extracted from convolutional neural networks and respectively train a ranking SVM and a classification network to predict region saliency. Kim and Pavlovic [46] use convolutional neural network as a multi-label classifier to predict the binary maps for region proposals. In [14] and [47], two convolutional neural networks are trained to respectively capture local and

global saliency cues of superpixels. Besides, recurrent convolutional networks are also applied in saliency detection [15], [48], [49]. Wang *et al.* [15] incorporate saliency prior knowledge into networks and automatically refine current results by using the contextual information in previous iterations. Kuen *et al.* [48] propose a recurrent attention convolutional-deconvolution network, which can locally refine the saliency map in the selected sub-regions in a progressive way. Liu and Han [49] use a hierarchical recurrent convolutional network to refine the details of saliency maps by integrating local context information step by step. In this work, instead of handcrafted features, we utilize FCN-32S [19] features due to its excellent performance in semantic segmentation.

III. Absorbing Markov Chain Model With Learnt Transition Probability

A. Principle of Absorbing Markov Chain

Given a set of states, $\mathcal{S} = \{s_1, s_2, \ldots, s_k\}$, the process starts in one of these states and moves to another. If the chain is currently in state s_i, it moves to state s_j at the next step with a probability denoted by p_{ij}. This probability is called transition probability, which does not depend upon which state the chain is in before the current state. In a Markov chain, it is allowed that the process moves from one state to itself with a probability denoted by p_{ii}. The state s_i is called transient state when $p_{ii} \neq 1$, while it is absorbing state when $p_{ii} = 1$, which means $p_{ij} = 0$ for all $i \neq j$. A Markov chain is completely specified by the transition probability matrix $\mathbf{P}$, which encodes the transition probability between any pair of states.

Considering an absorbing Markov chain with r absorbing states and n transient states, if we rearrange all the states so that transient ones are listed before absorbing states, the transition probability matrix $\mathbf{P}$ has the following canonical form,

$$\mathbf{P} \rightarrow \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}, \tag{1}$$

where $\mathbf{Q} \in \mathbb{R}^{n \times n}$ represents the transition probability matrix of transient states, while $\mathbf{R} \in \mathbb{R}^{n \times r}$ contains the probabilities of moving from any transient state to any absorbing one. $\mathbf{0}$ is the $r \times n$ zero matrix and $\mathbf{I}$ is the $r \times r$ identity matrix.

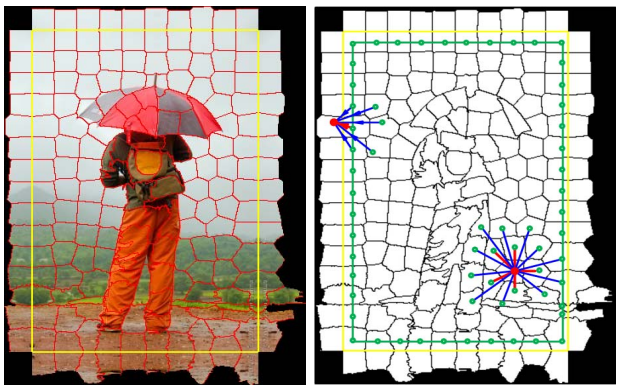
For an absorbing Markov chain, we can derive its fundamental matrix $\mathbf{F} = (\mathbf{I} - \mathbf{Q})^{-1} = \mathbf{I} + \mathbf{Q}^1 + \mathbf{Q}^2 + \mathbf{Q}^3 + \ldots$, where its element f_{ij} gives the expected number of times that the process has changed from transient state s_i to transient state s_j. And the sum $\sum_{j=1}^{n} f_{ij}$ reveals the expected number of times before absorption (into any absorbing state), which is also called absorbed time. Thus, we can compute the absorbed time for each transient state as

$$\mathbf{y} = \mathbf{F} \times \mathbf{1}_n, \tag{2}$$

where $\mathbf{1}_n$ is a n dimensional column vector all of whose elements are 1.

B. Saliency Model Based on Absorbing Markov Chain

For efficiency, the input image is first segmented into n superpixels via the simple linear iterative clustering (SLIC) approach in [50]. Then, we duplicate (mirror) the boundary superpixels along the border of image as shown in Figure 1(a). Next, an image graph is constructed to represent the absorbing Markov chain (AMC), in which superpixels are taken as nodes (or states). Since the boundary superpixels usually contain the global characteristics of the image background, we use the mirrored boundary superpixels as absorbing nodes and all other superpixels (inside the yellow bounding box) as transient nodes.



(a) (b)

Fig. 1. Absorbing Markov chain construction. (a) Absorbing and transient nodes. The superpixels outside the yellow bounding box are the mirrored boundary superpixels, which are regarded as absorbing nodes. While the superpixels inside the yellow bounding box are transient nodes. (b) Initial translation relationships. There are three kinds of relationships, which are respectively shown in red, blue and green colors. Noted that any pair of absorbing nodes are unconnected and any pair of transient nodes are symmetrically connected. While the connections from transient nodes to absorbing ones are unidirectional.

Given the graph affinity matrix $\mathbf{A}$, its element a_{ij} depicts the similarity between node s_i and s_j. The degree matrix that records the sum of the edge weights connected to each node is written as

$$\mathbf{D} = Diag(\sum\nolimits_j a_{ij}), \tag{3}$$

where $Diag(\cdot)$ is a matrix with its vector argument on the main diagonal. Thus, the transition probability matrix $\mathbf{P}$ of the absorbing Markov chain is computed as

$$\mathbf{P} = \mathbf{D}^{-1} \times \mathbf{A}, \tag{4}$$

which is actually the raw normalized $\mathbf{A}$.

Finally, according to Eq. (1) and Eq. (2), we can calculate the absorbed time $\mathbf{y}$ of all transient nodes (all n superpixels in the input image). By normalizing $\mathbf{y}$ to the range between 0 and 1, a saliency map S is obtained, that is

$$S(i) = \bar{\mathbf{y}}(i), \quad i = 1, 2, \ldots, n, \tag{5}$$

where i indexes the transient nodes and $\bar{\mathbf{y}}$ denotes the normalized absorbed time vector.

C. Learnt Transition Probability Matrix

In our previous work [17], we consider the local context information of each node (i.e. each node is only explicitly connected with its spatial neighbors), which is very important

for random walk, to construct a sparsely connected graph and obtain a sparse transition probability matrix. However, the sparse matrix restricts the random walk to move within a local region in each step, which is one of the reasons why the absorbed time is not effective when there are long-range smooth background regions near the image center. In this work, we learn a transition probability matrix by employing a *sparse-to-full* manner and combining multiple kinds of features.

The construction of learnt transition probability matrix is detailed as follows. Firstly, we construct a sparsely connected graph $G(V, E)$ with superpixels as nodes V and the links of pairs of nodes as edges E. We define the edges according to three criterions: (1) Each node is connected to its neighbouring nodes; (2) Each node is connected to ones which share common boundaries with its neighbouring nodes; (3) All nodes around the image borders are fully connected with each other, which can reduce the geodesic distance between similar superpixels. The specific connection relationship is shown in Figure 1(b). The weights of the edges encode the pairwise affinities, i.e, the nodes connected by an edge with large weight are considered to be strongly connected and have similar or even the same labels, while small weights represent nearly disconnected nodes and they have different or even opposite labels. The weight is defined as

$$\omega_{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma}}, \tag{6}$$

where $\mathbf{x}_i$ is the feature vector of node s_i. In this work, we utilize the pre-trained FCN-32S network [19] to extract the feature vector for each pixel and then average the vectors of all pixels in the node to represent the corresponding node. σ controls the strength of the weights, which is a constant in the experiments. We renumber the nodes so that the first n nodes are transient nodes and the last r nodes are absorbing nodes. Thus, we obtain the graph affinity matrix, which represents the correlation of any pair of nodes as

$$a_{ij} = \begin{cases} \omega_{ij}, & \text{if } j \in \mathcal{N}_i, 1 \le i \le n; \\ 1, & \text{if } i = j, n < i \le n + r; \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Here, $\mathcal{N}_i$ denotes the set of the nodes (including absorbing nodes) connected to transient node s_i. As the nodes are locally connected, the graph affinity matrix $\mathbf{A}$ is a sparse matrix with a small number of nonzero elements.

Then we learn to construct an approximatively full affinity matrix as follows:

$$\min_{\mathbf{W}} \sum_{i,j=1}^{n} a_{ij} \|\mathbf{w}_i - \mathbf{w}_j\|^2 + \mu \sum_{i=1}^{n} \|\mathbf{w}_i - \mathbf{i}_i\|^2, \tag{8}$$

where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n] \in \mathbb{R}^{n \times n}$ is the learnt affinity matrix based on the handcrafted sparse graph, in which $\mathbf{w}_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^\top$ is a column vector that denotes the similarities between node s_i and all other nodes. $\mathbf{i}_i$ is the i-th column of an identity matrix $\mathbf{I}$, which indicates that node s_i is only similar to itself initially. The smoothness constraint term (the first term) in Eq. (8) emphasizes that two nodes should have approximate similarity relationships with the others if they are strongly connected to each other. The fitting constraint term (the second term) emphasizes that no matter how we update $\mathbf{w}_i$ for node s_i, it should not differ too much from the initial value. The parameter μ controls the balance of two constraints.

To better depict the behavior of random walk and capture saliency cues, we explore the transition probability calculation in multiple kinds of feature spaces to obtain a learnt affinity matrix, which can synthetically consider the effects of different features. In this work, we adopt different feature layers extracted from the pre-trained FCN-32S network [19] to compute multiple sparse affinities and then infer a full affinity matrix as follows:

$$\begin{aligned} \min_{\boldsymbol{\beta}, \mathbf{W}} & \sum_{v=1}^{m} \beta^{(v)^\gamma} \sum_{i,j=1}^{n} a_{ij}^{(v)} \|\mathbf{w}_i - \mathbf{w}_j\|^2 \\ & + \mu \sum_{i=1}^{n} \|\mathbf{w}_i - \mathbf{i}_i\|^2, \\ s.t. & \sum_{v=1}^{m} \beta^{(v)} = 1, \quad 0 \le \beta^{(v)} \le 1, \end{aligned} \tag{9}$$

where m is the number of kinds of features and v indexes different features. $\boldsymbol{\beta}$ is a m dimensional column vector, whose element represents the importance of the corresponding affinity matrix. The parameter γ controls the weight distribution across multiple affinity matrices, which ensures that the complementary nature among multiple features is utilized. If we do not use parameter γ, the final solution of $\boldsymbol{\beta}$ satisfies that $\beta^{(v)} = 1$ for some type of feature v and $\beta = 0$ for the rest of features, which means only one kind of feature is effective.

For convenience, we rewrite Eq. (9) in matrix form, which is

$$\mathbf{J} = \sum_{v=1}^{M} \beta^{(v)^\gamma} Tr(\mathbf{W}^T \mathbf{L}^{(v)} \mathbf{W}) + \mu \|\mathbf{W} - \mathbf{I}\|_F^2. \tag{10}$$

$\mathbf{L}^{(v)} = \mathbf{D}^{(v)} - \mathbf{A}^{(v)}$ is the graph Laplacian matrix, where $\mathbf{D}^{(v)}$ and $\mathbf{A}^{(v)}$ are respectively the degree matrix and the graph affinity matrix computed by the v-th feature. $Tr(\cdot)$ and $\|\cdot\|_F$ calculate the trace and the Frobenius norm of the input matrix, respectively. We iteratively solve the optimization problem by decomposing it into two sub-problems: Fixing $\boldsymbol{\beta}$, updating $\mathbf{W}$ by

$$\mathbf{W} = \mu \Big(\sum_{v=1}^{m} \beta^{(v)^\gamma} \mathbf{L}^{(v)} + \mu \mathbf{I}\Big)^{-1}, \tag{11}$$

and fixing $\mathbf{W}$, updating $\boldsymbol{\beta}$ by

$$\beta^{(v)} = \frac{(Tr(\mathbf{W}^T \mathbf{L}^{(v)} \mathbf{W}))^{\frac{1}{1-\gamma}}}{\sum_{v'=1}^{m} (Tr(\mathbf{W}^T \mathbf{L}^{(v')} \mathbf{W}))^{\frac{1}{1-\gamma}}}. \tag{12}$$

More details about the solution process have been presented in [18]. After obtaining the learnt affinity matrix $\mathbf{W}_L$, we normalize it as in Eq. (4) to compute the learnt transition probability matrix $\mathbf{P}_L$ and saliency maps are computed based on $\mathbf{P}_L$. Figure 2 demonstrates several examples to visually compare

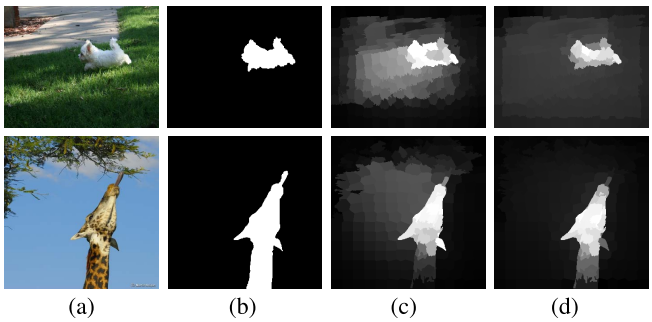


Fig. 2. Comparisons of traditional and learnt transition probability matrices. (a) Input images (b) Ground truth (c) Results produced by our previous work [17]. To be fair, we utilize the same two layers of deep features and concatenate them into a single feature vector. (d) Results generated by learnt transition probability matrix $\mathbf{P}_L$.

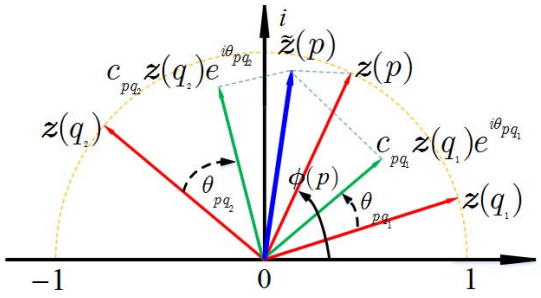


Fig. 3. Illustration of angular embedding. Given confidence matrix $\boldsymbol{C}$ and local ordering matrix $\boldsymbol{\Theta}$, $\tilde{z}(p)$ (the blue arrow) can be explained as the reconstruction result of $z(p)$ (the red arrow) based on matrix $\boldsymbol{C}$ and $\boldsymbol{\Theta}$. When the reconstruction error ε defined in Eq. (13) is minimized, $z(p) = e^{i\boldsymbol{\phi}(p)}$ is an embedding of node s_p into the complex plane and angle $\boldsymbol{\phi}(p)$ is the global ordering of node s_p.

the results produced by the traditional transition probability matrix in [17] and the learnt transition probability matrix $\mathbf{P}_L$, from which we can see that the latter is more effective.

IV. Refinement Model

Although the detection results are improved by the learnt transition probability matrix, there still exists the drawback that foreground regions are not uniformly highlighted as shown in Figure 2(d). To address this issue, we propose a novel refinement model based on angular embedding, which provides a mathematical framework for solving perceptual figure/ground organization problems.

A. Principle of Angular Embedding

Angular embedding (AE) is a novel ranking principle, whose goal is to find the global orderings of all elements where their relative differences between elements match well the pairwise local ordering measurements. The solution process is similar to spectral partitioning algorithm. The difference is that AE needs a pair of real-valued matrices $(\boldsymbol{\Theta}; \boldsymbol{C})$, where $\boldsymbol{\Theta}$ is called local ordering matrix which encodes pairwise local orderings and $\boldsymbol{C}$ is a confidence matrix which indicates the confidence level of each local ordering. The AE produces the complex-valued eigenvectors and their angles figure the global ordering relationships.

Assuming that the local ordering $\theta_{pq} \in \boldsymbol{\Theta}$ between some pairs of nodes s_p and s_q are given, the AE realizes the global ordering $\boldsymbol{\phi}$ that agrees as best as possible with the given local orderings in the angles of nodes on a unit circle. The process of mapping is shown in Figure 3. In [51]–[53], this optimization problem is addressed by minimizing error ε,

$$\varepsilon = \sum_p \frac{\sum_q c_{pq}}{\sum_{p,q} c_{pq}} \cdot |z(p) - \tilde{z}(p)|^2, \tag{13}$$

where $c_{pq} \in \boldsymbol{C}$ encodes the confidence of the local ordering between node s_p and node s_q. $z(p) = e^{i\boldsymbol{\phi}(p)}$ is an embedding of node s_p into the complex plane and $\boldsymbol{\phi}(p)$ is the global ordering. And $\tilde{z}(p)$ can be regarded as the reconstruction result of $z(p)$ according to its neighbors and their local orderings $\boldsymbol{\Theta} = [\theta_{pq}]_{n\times n}$, which is defined as

$$\tilde{z}(p) = \sum_q \tilde{c}_{pq} \cdot e^{i\theta_{pq}} \cdot z(q), \tag{14}$$

where

$$\tilde{c}_{pq} = \frac{c_{pq}}{\sum_q c_{pq}}. \tag{15}$$

Rewriting Eq. (13) in matrix form and relaxing the unit norm constraint on $z(\cdot)$ yield a generalized eigenproblem,

$$\mathbf{K}z = \lambda \mathbf{D}z, \tag{16}$$

where

$$\mathbf{K} = (\mathbf{I} - \mathbf{M})^* \mathbf{D} (\mathbf{I} - \mathbf{M}), \tag{17}$$

$$\mathbf{M} = Diag(\boldsymbol{C}\mathbf{1}_n)^{-1} \boldsymbol{C} \circ e^{i\boldsymbol{\Theta}}, \tag{18}$$

$$\mathbf{D} = Diag(\boldsymbol{C}\mathbf{1}_n), \tag{19}$$

where $\boldsymbol{C} = [c_{pq}]_{n\times n}$, the symbol $*$ denotes complex conjugate transpose and $\circ$ is the matrix Hadamard product. $Diag(\cdot)$ is a matrix with its vector argument on the main diagonal, and $\mathbf{1}_n$ is a n dimensional column vector of ones.

The global ordering is encoded in the angle of the first eigenvector z_0 corresponding to minimum eigenvalue, which is computed by

$$\boldsymbol{\phi} = \measuredangle z_0. \tag{20}$$

B. Refinement Model Based on AE

As introduced above, angular embedding need local ordering matrix $\boldsymbol{\Theta}$ and confidence matrix $\boldsymbol{C}$ as a precondition. In this work, according to saliency maps produced by AMC and boundary maps generated by [20], we determine the local ordering between each pair of superpixels and rearrange saliency values of all superpixels, thereby uniformly highlighting the whole salient object and elevating the contrast between salient regions and backgrounds.

Matrix $\boldsymbol{\Theta}$ encodes the local orderings, whose element θ_{pq} expresses the pairwise ordering relationship of saliency values between any pair of node s_p and s_q. Specifically, if s_p belongs to salient regions and s_q belongs to the background (i.e., the saliency value of s_p is larger than that of s_q), θ_{pq} is

a positive constant and θ_{qp} is a negative constant, which indicate the inter-class (between salient object class and background class) local orderings. If both s_p and s_q belong to salient (or background) regions, θ_{pq} and θ_{qp} are set to 0, which indicate the intra-class local orderings. To define $\boldsymbol{\Theta}$, we first calculate the saliency differences of all pairs of nodes, and obtain a matrix $\mathbf{B} = [b_{pq}]_{n\times n}$ as follows:

$$b_{pq} = S(p) - S(q), \tag{21}$$

where p and q indexes all the superpixels, and $S(p)$ is the saliency value of s_p computed by Eq. (5) and $\mathbf{P}_L$. The absolute value $|b_{pq}|$ denotes the validity of the inter-class local ordering. The large $|b_{pq}|$ means that the two nodes very possibly come from the foreground and background regions, respectively, which possibly yields an inter-class local ordering. While the small $|b_{pq}|$ signifies that the two nodes possibly belong to the same homogeneous region, which might generate an intra-class local ordering.

Because the accuracy of local orderings directly affects the global ordering results of AE, we propose two criterions to precisely determine inter-class and intra-class local orderings: (1) Saliency contrast, i.e., the saliency difference between two nodes should be larger than a fixed threshold value τ; (2) Boundary contrast, i.e., two nodes cannot be in the same homogeneous region, which means that there exists strong boundaries between the two nodes. The saliency contrast is determined by the saliency maps produced by AMC in Section III. The maps are not necessarily always accurate. Therefore, the boundary contrast is proposed to avoid their negative effects on the identification of local orderings. It depends on boundary maps produced by [20], and we exploit e_{pq} to denote the maximum value of boundary strength along the line connecting node s_p and s_q. Specifically, the two criterions are defined as :

$$\pi^1_{pq} = \begin{cases} 1, & \text{if } |b_{pq}| > \tau; \\ 0, & \text{otherwise,} \end{cases} \tag{22}$$

and

$$\pi^2_{pq} = \begin{cases} 1, & \text{if } e_{pq} > 0; \\ 0, & \text{otherwise,} \end{cases} \tag{23}$$

where $\Pi^1 = [\pi^1_{pq}]_{n\times n}$ and $\Pi^2 = [\pi^2_{pq}]_{n\times n}$ are two indicator matrices. If a local ordering meets both criterions, it is considered to be inter-class, otherwise it is intra-class. Finally, the local ordering matrix $\boldsymbol{\Theta}$ is computed by

$$\boldsymbol{\Theta} = \varphi \cdot sign(\mathbf{B} \circ \Pi^1 \circ \Pi^2), \tag{24}$$

where the symbol $\circ$ denotes the matrix Hadamard product and $sign(\cdot)$ is a sign function:

$$sign(x) = \begin{cases} 1, & \text{if } x > 0; \\ -1, & \text{if } x < 0; \\ 0, & \text{otherwise.} \end{cases} \tag{25}$$

The parameter φ is a positive constant, that is, the absolute values of all the inter-class local orderings are the same, which helps reduce the hierarchy of global ordering, thereby suppressing mid-level saliency. Figure 4 illustrates the process of determining the inter-class local orderings.

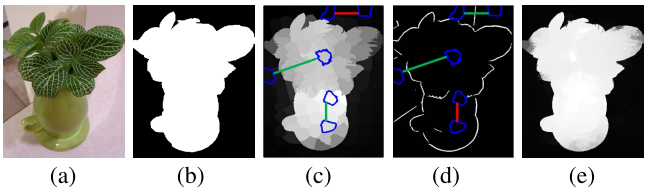


Fig. 4. Pairwise local orderings. From left to right: (a) Input image (b) Ground truth (c) Saliency map generated by AMC with learnt transition probability matrix. Note that the blue closed curves represent different superpixel nodes. The green (or red) lines denote some pairwise local orderings between two nodes, where the green lines indicate the local orderings that meet saliency contrast constraint (i.e., two nodes have large saliency difference), while the red line means the local ordering does not meet this constraint. (d) Binary boundary map produced by [20]. The green lines indicate the local orderings that meet boundary contrast constraint (i.e., two nodes straddle strong image boundaries), while the red line means the local ordering does not meet this constraint. (e) Final saliency map refined by the AE. If and only if both criterions are met, the corresponding local orderings are determined to be inter-class local orderings, otherwise they belong to intra-class local orderings.

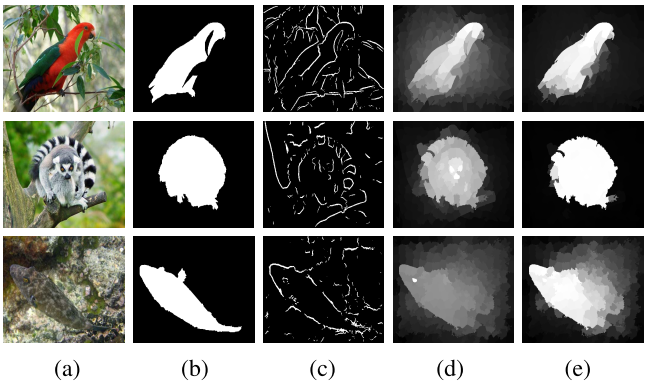


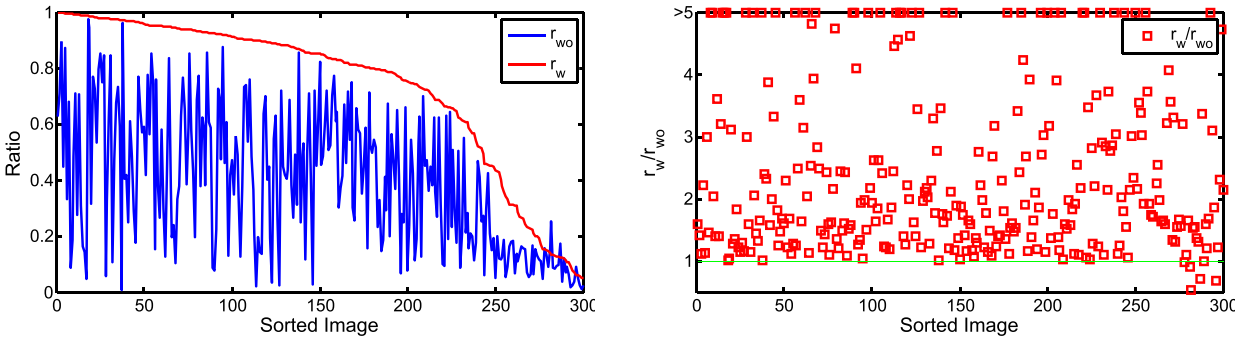
Fig. 5. The effect of refinement. (a) Input images. (b) Ground truth. (c) Binary boundary maps. (d) Results without refinement. (e) Results refined by angular embedding.

Confidence matrix $\boldsymbol{C}$ can be taken as the strength of connection between nodes. Two nodes in the perceptually homogeneous region are usually strongly connected. Thus, they have high confidence c_{pq}, which means that their local ordering cannot change too much in the global ordering. While two nodes in different regions (e.g., figure and ground) are weakly connected and have low confidence, they will be optimally re-ranked because local ordering θ_{pq} only regulates pairwise local ordering of saliency and does not encode their relative strength of saliency. In this work, we directly employ learnt affinity matrix $\mathbf{W}_L$ as $\boldsymbol{C}$, which can reduce calculation time greatly and perform better than traditional affinity matrix.

After obtaining $(\boldsymbol{\Theta}; \boldsymbol{C})$, we compute the first eigenvector z_0 by Eq. (16)-(19), and then normalize the angle $\boldsymbol{\phi}$ of z_0 to the range between 0 and 1, thereby generating the refined saliency map S_r, that is,

$$S_r(p) = \bar{\boldsymbol{\phi}}(p), \quad p = 1, 2, \ldots, n, \tag{26}$$

where p indexes the superpixels and $\bar{\boldsymbol{\phi}}$ denotes the normalized angle vector. As shown in Figure 5, salient objects are



(a) (b)

Fig. 6. The effect of AE. (a) The horizontal axis denotes different images, which are sorted by r_w value. The vertical axis means the ratio of the overall number of salient (> 0.9) and background (< 0.1) pixels to the total number of pixels in the saliency map. r_w and r_{wo} respectively denote the ratio for the map with and without refinement. (b) $r_w/r_{wo} > 1$ means that the contrast between foreground and background in the map is enhanced by AE.

uniformly highlighted and the contrast between foreground and background is significantly boosted by the AE refinement. In addition, this figure also demonstrates the effects of the two criterions. When the boundary contrast is pretty weak, the AE model can still achieve good results (the first two examples). Similarly, when the saliency contrast is not good, salient object can also be further highlighted with the help of the boundary contrast (the last example). In other words, the two criterions are complementary to each other.

To statistically analyze the effect of the AE, we randomly select 300 images from the ECSSD dataset, and then compute the ratio $r\%$ of the overall number of salient (saliency value > 0.9) and background (saliency value < 0.1) pixels to the total number of pixels in the saliency map for each image. This ratio is able to proximately describe the binarization degree of detection results (the ratio is equal to 1 for ground truth). Thus, given a saliency map, we consider that $r\%$ of pixels can be reliably decided to belong to salient regions or background, while $1 - r\%$ of pixels cannot be definitely accurately assigned their class labels, especially for mid-level saliency pixels. Let r_w and r_{wo} respectively denote the ratio for the map with and without refinement. The result is shown in Figure 6(a), where we re-sort images according to r_w. We can see that the curve of r_w (red color) is above that of r_{wo} (blue color), which indicates that the regions with mid-level saliency are decreased by AE. To better explain it, we also compute the ratio of r_w/r_{wo} as shown in Figure 6(b). r_w/r_{wo} is much larger than one for most images, which means that the contrast between foreground and background in the map is significantly enhanced. The main steps of the proposed saliency detection algorithm are summarized in Algorithm 1.

Algorithm 1 Saliency Detection Based on Absorbing Markov Chain

Input: An image, a boundary map corresponding to the image and required parameters.

- AMC with learnt transition probability matrix:

1. Construct a sparsely connected graph $G(V, E)$ with superpixels as nodes, and use boundary nodes as absorbing nodes.
2. Compute ν-th graph affinity matrix $\mathbf{A}^{(\nu)}$ with different features by Eq. (7).
3. Compute degree matrix $\mathbf{D}^{(\nu)}$ by Eq. (3) and ν-th graph Laplacian matrix by $\mathbf{L}^{(\nu)} = \mathbf{D}^{(\nu)} - \mathbf{A}^{(\nu)}$.
4. Initialize $\boldsymbol{\beta} = 1/m$ (m represents the number of kinds of features).
5. **while** not convergent or not reaching the maximum iterations **do**
6. fix $\boldsymbol{\beta}$, optimize affinity matrix $\mathbf{W}$ by Eq. (11).
7. fix $\mathbf{W}$, optimize weight $\boldsymbol{\beta}$ by Eq. (12).
8. **end while**

(*Note that after all iterations, learnt affinity matrix $\mathbf{W}_L$ is achieved.*)

9. Compute learnt transition probability matrix $\mathbf{P}_L$ by normalizing $\mathbf{W}_L$.
10. Extract the matrix $\mathbf{Q}$ from $\mathbf{P}_L$ by Eq. (1), and compute the fundamental matrix by $\mathbf{F} = (\mathbf{I} - \mathbf{Q})^{-1}$.
11. Compute the saliency map S by Eq. (2) and Eq. (5).

- Refined by AE:

12. Compute matrix $\mathbf{B}$ by Eq. (21).
13. Compute indicator matrices $\mathbf{\Pi}^1$ and $\mathbf{\Pi}^2$ by Eq. (22) and Eq. (23).
14. Calculate local ordering matrix $\boldsymbol{\Theta}$ by Eq. (24) and set confidence matrix $\boldsymbol{C} = \mathbf{W}_L$.
15. Compute the first eigenvector $\boldsymbol{z}_0$ by Eq. (16)-(19).
16. Obtain the angle $\boldsymbol{\phi}$ of $\boldsymbol{z}_0$ by Eq. (20) and refine saliency map S_r by Eq. (26).

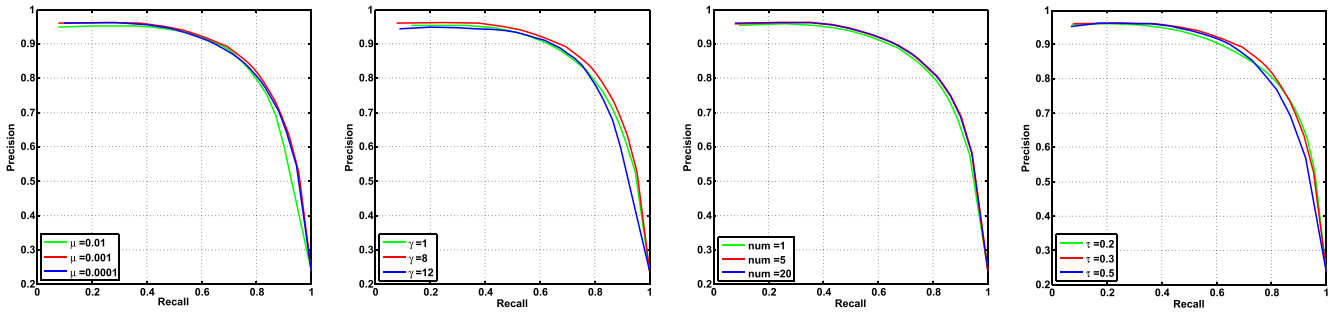
Output: The refined saliency map.

V. Experiments

We evaluate the proposed algorithm on seven common benchmark datasets, and compare it with thirteen state-of-the-art saliency methods, including the BL [11], MR [8], GC [7], HDCT [54], HS [55], wCtr [56], PCA [25], RR [42], UFO [5], MAP [43], DRFI [13], LEGS [14] and KSR [44]. Among them, DRFI [13], LEGS [14] and KSR [44] require more than $2,000$ annotated images from saliency datasets to train their parameter models. In the following, we detail the datasets, parameter settings, evaluation criterions, examinations of design options, in which we compare the proposed approach against our previous work [17], comparisons with the state-of-the-art models, run time and failure cases.

A. Datasets

We use the ECSSD, PASCAL-S, SOD, MSRA, ASD, HKU-IS and SED1 datasets and all datasets provide accurate human-labelled pixel-wise ground truth. The ECSSD dataset [55] contains $1,000$ structurally complex images. The PASCAL-S dataset [57] is composed of 850 natural images, which is one of the most challenging saliency datasets. The SOD dataset [58] is another challenging dataset,



(a) (b) (c) (d)

Fig. 7. Precision-recall curves on the ECSSD dataset by the proposed algorithm with different parameter values. (a) Results with different values of μ in Eq. (9). (b) Results with different values of γ in Eq. (9). (c) Results with different numbers of iteration. (d) Results with different values of τ in Eq. (22).

which includes 300 images from the Berkeley segmentation dataset. Some images in this dataset comprise multiple salient object instances with different sizes. The MSRA dataset contains 5, 000 images with various image contents, which is first constructed by Liu *et al.* [59] with the ground truth marked by bounding boxes. Afterwards, Jiang *et al.* [13] provide accurate pixel-wise ground truth. The ASD dataset, a subset of the MSRA dataset, contains 1,000 images with accurate human-labelled ground truth. The HKU-IS dataset [45] contains 4, 447 images which are divided into training and testing parts. The SED1 dataset [60] consists of 100 images with objects of largely different sizes and locations.

B. Parameter Settings

In the process of constructing the sparsely connected graph model, there are two parameters n and σ. n represents the number of superpixels, which is fixed to 250 in all experiments. σ in Eq. (6) controls the strength of weight between a pair of nodes and is set to be $\sigma^2 = 0.1$.

To compute the learnt transition probability matrix, we utilize FCN-32S [19] features due to its great success in semantic segmentation, and only choose the outputs of *Conv1* and *Conv5* layers as feature maps, which means the parameter m in Eq. (9) is fixed to be 2. This is because the features in the *Conv1* of CNNs encode the low-level detailed information and the features in the *Conv5* carry the higher-level semantic abstraction, which are of vital importance for detecting salient objects. We resize the feature maps of both layers to the input size via bilinear interpolation to preserve the same resolution between feature maps and input image. In this way, two different kinds of pixel-wise features are obtained, which has 64 and 512 dimensions, respectively. In Eq. (9), the parameter μ and γ are respectively set to 0.001 and 8, which are empirically determined. We find that the detection results are insensitive to these two parameters as shown in Figure 7(a) and (b). The parameter $\boldsymbol{\beta}$ is a 2 dimensional vector, whose elements represent the importance of the corresponding features. It is initialized with $[0.5, 0.5]^\top$ and adaptively updated by Eq. (12). Figure 7(c) demonstrates the results of different numbers of iteration when solving the optimization formula in Eq. (9), which is finally fixed to 5 in all experiments.

In the process of refinement, the threshold value τ in Eq. (22) is set to 0.3. Similarly, our saliency results are insensitive to τ when it is inside the range between 0.2 and 0.5 as shown in Figure 7(d). All the foreground-background local orderings are set to a positive constant φ in matrix $\boldsymbol{\Theta}$. In fact, we only need guarantee that the parameter φ is positive regardless of its numerical value, because the sum of all local ordering values will be normalized in the end.

C. Evaluation Criterions

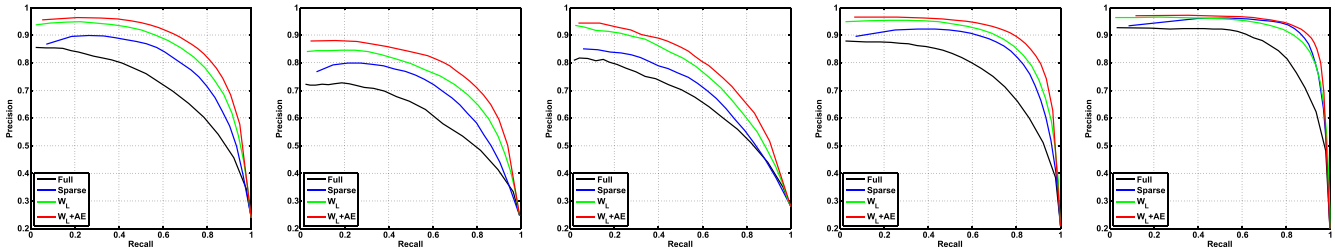
Precision and recall (P-R) curve, F-measure and Area Under Curve (AUC) are three common quantitative criterions in saliency evaluation. The precision is defined as the ratio of correctly retrieved salient pixels to all the pixels of extracted regions, while recall corresponds to the ratio of correctly retrieved salient pixels to all salient pixels in ground truth. P-R curve is a scatterplot with smooth lines, where the precision and recall are respectively the y-value and x-value. We segment saliency map using every threshold in the range [0:0.05:1], and then compute pairs of precision and recall at each threshold value to plot the P-R curve. The F-measure is the overall performance measurement computed by the weighted harmonic of precision and recall. We first compute the precision and recall with an adaptive threshold proposed in [61], which is defined as twice the mean saliency of the image. Then, the F-measure is defined as follows:

$$F_\xi = \frac{(1+\xi^2) \times Precision \times Recall}{\xi^2 \times Precision + Recall}, \tag{27}$$

where ξ^2 is set to 0.3 to emphasize the precision. Meanwhile, we obtain true positive and false positive rates to plot the ROC curve and compute the area under the ROC curve (AUC score).

D. Examinations of Design Options

We examine each step of the proposed algorithm on five datasets, including the ECSSD, PASCAL-S, SOD, MSRA and ASD dataset. The results are shown in Figure 8, in which the green curves are the results without refinement, that is, we employ two layers of FCN features to jointly learn and combine two dense affinity matrices by Eq. (10), thereby generating a learnt transition probability matrix. While the



(a) (b) (c) (d) (e)

Fig. 8. Precision-recall curves of the proposed algorithm with different design options on five datasets. The black curves are the results of handcrafted full affinity matrix, while the blue curves are the results of sparse affinity matrix [17]. To be fair, we concatenate the same two kinds of features into a single feature vector. The green curves are the results of learnt transition probability matrix. The results refined by angular embedding are shown in red curves, which makes a contribution to the performance. (a) ECSSD. (b) PASCAL. (c) SOD. (d) MSRA. (e) ASD.

TABLE I

QUANTITATIVE COMPARISONS IN TERMS OF F-MEASURE SCORE. THE BEST AND SECOND BEST RESULTS ARE SHOWN IN RED COLOR AND BLUE COLOR RESPECTIVELY

Datasets	GC	PCA	HS	UFO	RR	wCtr	MAP	BL	HDCT	MR	LEGS	DRFI	KSR	ours
ECSSD	0.5726	0.5796	0.6363	0.6442	0.6577	0.6774	0.7005	0.6825	0.6897	0.6932	0.7852	0.7337	0.7705	0.8004
PASCAL	0.4861	0.5298	0.5278	0.5502	0.5873	0.5972	0.5864	0.5668	0.5824	0.5881	0.6951	0.6159	0.6760	0.6952
SOD	0.4642	0.5370	0.5210	0.5480	0.5665	0.5978	0.5828	0.5723	0.6108	0.5697	0.6492	0.6031	0.6622	0.6578
MSRA	0.6575	0.6723	0.7115	0.7265	0.7575	0.7437	0.7891	0.7328	0.7364	0.7510	-	-	0.7763	0.8567
HKU-IS	-	0.5778	0.6358	-	0.6672	0.6769	0.6585	0.6597	0.6584	0.6549	0.7228	0.7218	0.7465	0.7840
SED1	-	0.6459	0.7221	-	0.8217	0.7848	0.8175	0.7809	0.7692	0.8108	0.8522	0.8133	0.8030	0.8248

TABLE II

QUANTITATIVE COMPARISONS IN TERMS OF AUC SCORE. THE BEST AND SECOND BEST RESULTS ARE SHOWN IN RED COLOR AND BLUE COLOR RESPECTIVELY

Datasets	GC	PCA	HS	UFO	RR	wCtr	MAP	BL	HDCT	MR	LEGS	DRFI	KSR	ours
ECSSD	0.7848	0.8737	0.8821	0.8587	0.8283	0.8779	0.8956	0.9147	0.9039	0.8820	0.9239	0.9391	0.9257	0.9264
PASCAL	0.7321	0.8371	0.8330	0.8088	0.8251	0.8433	0.8366	0.8633	0.8582	0.8205	0.8857	0.8913	0.8970	0.8674
SOD	0.7046	0.8212	0.8145	0.7840	0.7888	0.8014	0.8147	0.8503	0.8504	0.7899	0.8117	0.8464	0.8510	0.8423
MSRA	0.8398	0.9248	0.9043	0.8950	0.9089	0.9169	0.9370	0.9360	0.9318	0.9044	-	-	0.9376	0.9566
HKU-IS	-	0.8898	0.8782	-	0.8711	0.8951	0.8854	0.9140	0.8893	0.8611	0.9026	0.9435	0.9097	0.9259
SED1	-	0.9032	0.9038	-	0.9062	0.8985	0.9409	0.9463	0.9327	0.9003	0.9150	0.9617	0.9179	0.9217

black and the blue curves are respectively the results of handcrafted full affinity matrix and sparse affinity matrix (our previous work [17]). For a fair comparison, we concatenate the same two layers of deep features (the dimensions are respectively 64 and 512) into a 576 dimensional feature vector. Obviously, the learnt transition probability matrix is very effective for performance improvement. Besides, the AE refinement method can significantly enhance the overall saliency of the foreground objects and adequately suppress the backgrounds. The performances after refinement are demonstrated with the red curves in Figure 8, which are better than the results without refinement.

E. Comparisons With Other Models

Figure 9 shows the precision-recall curves and F-measure of different methods on six datasets. To be fair, we do not show the results of the DRFI [13] and LEGS [14] on the MSRA dataset, because the two methods randomly select more than $2,000$ images from this dataset to train their models. In addition, since the GC [7] and UFO [5] based detection results on the SED1 and HKU-IS datasets are not provided, we do not present the corresponding performance. As shown in Figure 9, we can see that the precision-recall curves of the proposed algorithm significantly perform better than other competitors on the ECSSD, SOD, MSRA and HKU-IS datasets. On the PASCAL dataset, we achieve comparable performance with the KSR [44] and LEGS [14], and significantly exceeds all other methods in terms of precision-recall curve. While on the SED1 dataset, the LEGS [14] and the proposed method perform equally well. In terms of precision, the proposed algorithm consistently performs better than other competitors on six datasets.

The F-measure scores of different methods are shown in Table I. We achieve the highest F-measure values on the ECSSD, PASCAL, MSRA and HKU-IS datasets and perform second-best on the SOD and SED1 datasets, slightly worse than the KSR [44] and LEGS [14], respectively. Table II demonstrates the AUC scores of all evaluated methods. We can see a performance drop of our approach. The AUC is computed by the area under the ROC curve and the ordinate of ROC curve is true positive rate, which is identical to the recall. Because our method is inclined to achieve higher precision and suppress the recall, it yields relatively low AUC score. However, the proposed algorithm overall performs comparable with the DRFI [13] and KSR [44] on six datasets, which are two supervised methods as mentioned above. What is more, Figure 10 shows a few saliency maps generated by the evaluated methods. We can find that the proposed algorithm uniformly highlights the salient regions with well-defined contours.

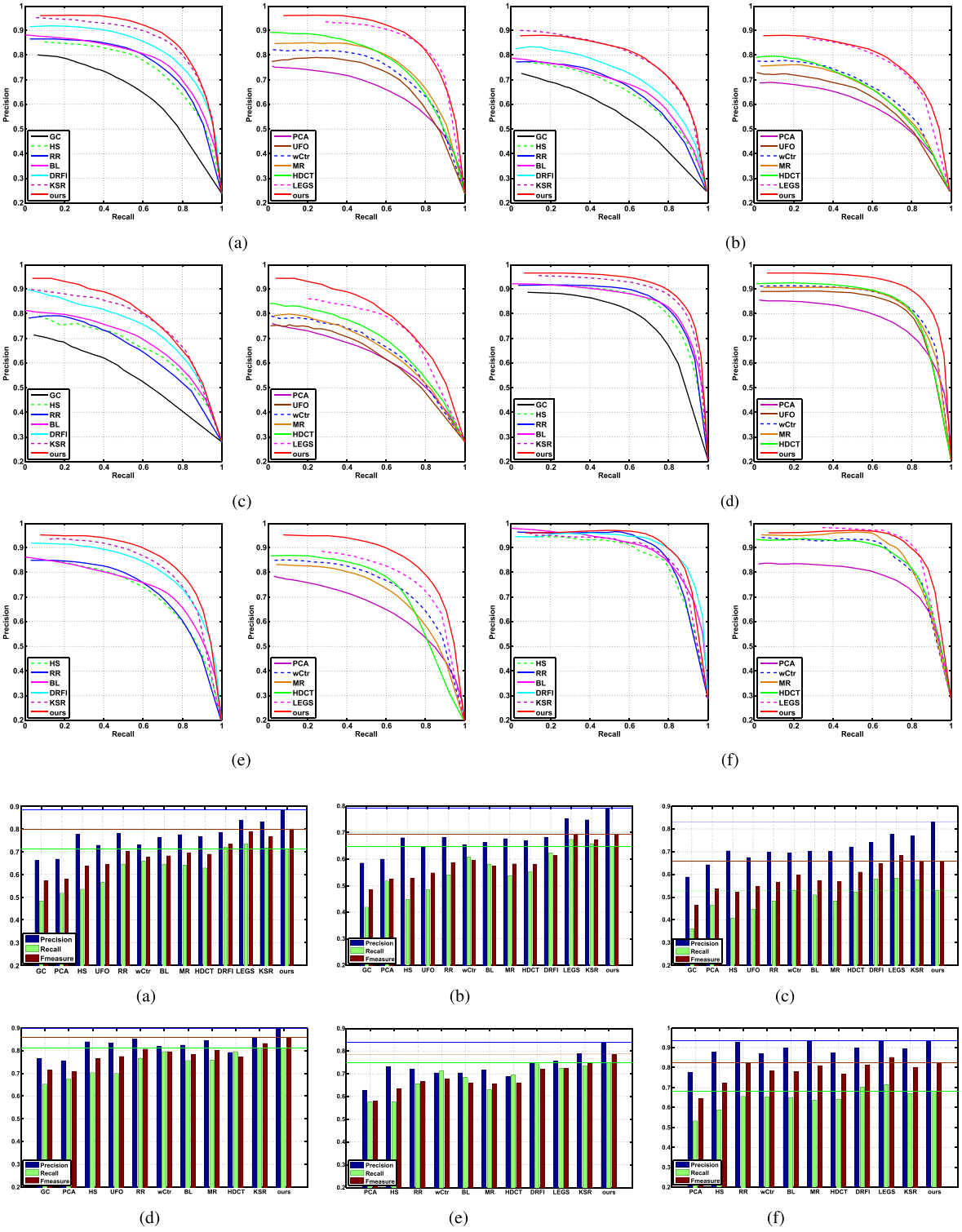


Fig. 9. Quantitative comparisons of different methods on six datasets. (a) ECSSD dataset. (b) PASCAL dataset. (c) SOD dataset. (d) MSRA dataset. (e) HKU-IS dataset. (f) SED1 dataset. (a) ECSSD dataset. (b) PASCAL dataset. (c) SOD dataset. (d) MSRA dataset. (e) HKU-IS dataset. (f) SED1 dataset.

F. Run Time

Without considering the computational cost of extracting deep features, the proposed algorithm takes on average 1.143s to process one image of 300×400 size via MATLAB implement on a machine equipped with an Intel i5-6500 3.20GHz CPU and 8GB RAM. The MATLAB code of the proposed algorithm will be made available to the public.

G. Failure Cases

In this work, we learn a transition probability matrix by using the *sparse-to-full* method and the discriminative FCN features. The proposed algorithm is highly effective for most tasks of saliency detection. However, when salient objects have similar appearances with the background and the images are filled with the cluttered background, our algorithm cannot

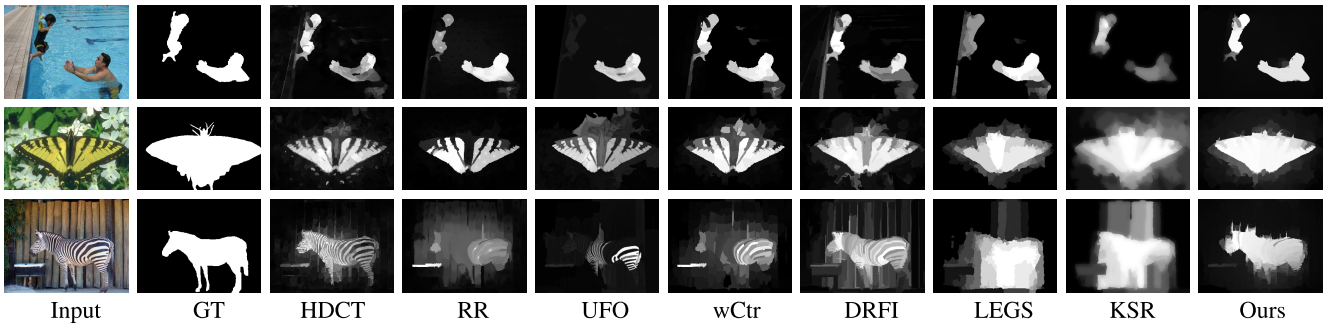


Fig. 10. Visual comparisons with seven state-of-the-art methods.

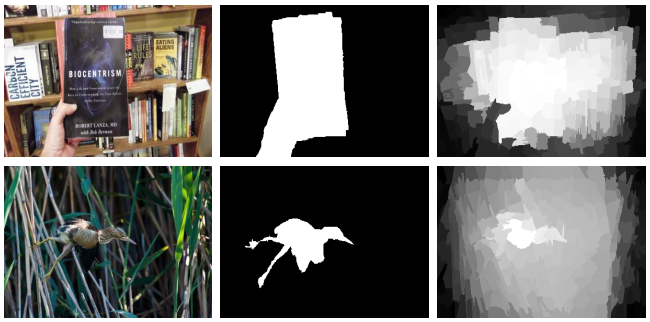


Fig. 11. Failure cases. (a) Input images. (b) Ground truth. (c) Saliency maps.

make the entire salient object be highlighted homogeneously as shown in Figure 11.

VI. Conclusion

In this paper, we propose a bottom-up saliency detection algorithm via absorbing Markov chain with learnt transition probability matrix. Based on boundary prior, we set the virtual boundary nodes as absorbing nodes. The absorbed time of each node is computed as its saliency. In the process of random walks, the transition probability matrix is very important for absorbed time computation. Therefore, we formulate transition matrix generation as an optimization problem by the *sparse-to-full* method combined with multiple-layer deep features. Actually, it can be also regarded as feature selection and integration process. Moreover, angular embedding technique is examined to refine saliency results, which can promote that salient objects are uniformly highlighted. Experimental results demonstrate that the proposed method performs favorably against thirteen state-of-the-art methods on six public datasets.

References

[1] S. Matsuo and K. Yanai, "CNN-based style vector for style image retrieval," in *Proc. ACM Int. Conf. Multimedia Retr.*, 2016, pp. 309–312.

[2] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, 2014.

[3] Z. Huang, R. Wang, S. Shan, and X. Chen, "Learning Euclidean-to-Riemannian metric for point-to-set classification," in *Proc. CVPR*, 2014, pp. 1677–1684.

[4] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?" in *Proc. CVPR*, 2004, pp. 37–44.

[5] P. Jiang, H. Ling, J. Yu, and J. Peng, "Salient region detection by ufo: Uniqueness, focusness and objectness," in *Proc. ICCV*, 2013, pp. 1976–1983.

[6] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proc. CVPR*, 2011, pp. 409–416.

[7] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook, "Efficient salient region detection with soft image abstraction," in *Proc. ICCV*, 2013, pp. 1529–1536.

[8] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. CVPR*, 2013, pp. 3166–3173.

[9] W.-C. Tu, S. He, Q. Yang, and S.-Y. Chien, "Real-time salient object detection with a minimum spanning tree," in *Proc. CVPR*, 2016.

[10] C. Scharfenberger, A. Wong, K. Fergani, and J. S. Zelek, "Statistical textural distinctiveness for salient region detection in natural images," in *Proc. CVPR*, Jun. 2013, pp. 979–986.

[11] N. Tong, H. Lu, X. Ruan, and M.-H. Yang, "Salient object detection via bootstrap learning," in *Proc. CVPR*, 2015, pp. 1884–1892.

[12] M. Maire, S. X. Yu, and P. Perona, "Object detection and segmentation from joint embedding of parts and pixels," in *Proc. ICCV*, 2011, pp. 2142–2149.

[13] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *Proc. CVPR*, 2013, pp. 2083–2090.

[14] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. CVPR*, 2015, pp. 3183–3192.

[15] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Proc. ECCV*, 2016, pp. 825–841.

[16] Z. Jiang and L. S. Davis, "Submodular salient region detection," in *Proc. CVPR*, 2013, pp. 2043–2050.

[17] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency detection via absorbing markov chain," in *Proc. ICCV*, 2013, pp. 1665–1672.

[18] S. Bai, S. Sun, X. Bai, Z. Zhang, and Q. Tian, "Smooth neighborhood structure mining on multiple affinity graphs with applications to context-sensitive similarity," in *Proc. ECCV*, 2016, pp. 592–608

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. CVPR*, 2015, pp. 3431–3440.

[20] C. L. Zitnick and P. Dollár "Edge boxes: Locating object proposals from edges," in *Proc. ECCV*, 2014, pp. 391–405.

[21] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[22] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. CVPR*, 2012, pp. 733–740.

[23] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai, "Fusing generic objectness and visual saliency for salient object detection," in *Proc. ICCV*, 2011, pp. 914–921.

[24] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" in *Proc. CVPR*, 2010, pp. 73–80.

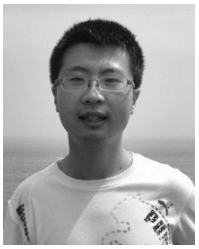
[25] R. Margolin, A. Tal, and L. Zelnik-Manor, "What makes a patch distinct?" in *Proc. CVPR*, 2013, pp. 1139–1146.

[26] Y. Xie, H. Lu, and M.-H. Yang, "Bayesian saliency via low and mid level cues," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1689–1698, May 2013.

[27] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proc. CVPR*, 2015, pp. 110–119.
[28] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Proc. CVPR*, 2012, pp. 853–860.
[29] C. Lang, G. Liu, J. Yu, and S. Yan, "Saliency detection by multitask sparsity pursuit," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1327–1338, Mar. 2012.
[30] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang, "Saliency detection via dense and sparse reconstruction," in *Proc. ICCV*, 2013, pp. 2976–2983.
[31] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," in *Proc. ECCV*, 2012, pp. 29–42.
[32] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Minimum barrier salient object detection at 80 fps," in *Proc. ICCV*, 2015, pp. 1404–1412.
[33] M. Long and F. Liu, "Comparing salient object detection results without ground truth," in *Proc. ECCV*, 2014, pp. 76–91.
[34] L. Zheng, C. Yang, H. Lu, X. Ruan, and M.-H. Yang, "Ranking saliency," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 9, pp. 1892–1904, Sep. 2017.
[35] Q. Wang, W. Zheng, and R. Piramuthu, "Grab: Visual saliency via novel graph model and background priors," in *Proc. CVPR*, 2016, pp. 535–543.
[36] C. Gong *et al.*, "Saliency propagation from simple to difficult," in *Proc. CVPR*, 2015, pp. 2531–2539.
[37] P. Jiang, N. Vasconcelos, and J. Peng, "Generic promotion of diffusion-based salient object detection," in *Proc. ICCV*, 2015, pp. 217–225.
[38] L. D. F. Costa. (Mar. 2006). "Visual saliency and attention as random walks on complex networks." [Online]. Available: https://arxiv.org/abs/physics/0603025
[39] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proc. NIPS*, 2006, pp. 545–552.
[40] V. Gopalakrishnan, Y. Hu, and D. Rajan, "Random walks on graphs for salient object detection in images," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3232–3242, Dec. 2010.
[41] J.-S. Kim, J.-Y. Sim, and C.-S. Kim, "Multiscale saliency detection using random walk with restart," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 2, pp. 198–210, Feb. 2014.
[42] C. Li, Y. Yuan, W. Cai, Y. Xia, and D. D. Feng, "Robust saliency detection via regularized random walks ranking," in *Proc. CVPR*, 2015, pp. 2710–2717.
[43] J. Sun, H. Lu, and X. Liu, "Saliency region detection based on Markov absorption probabilities," *IEEE Trans. Image Process.*, vol. 24, no. 5, pp. 1639–1649, May 2015.
[44] T. Wang, L. Zhang, H. Lu, C. Sun, and J. Qi, "Kernelized subspace ranking for saliency detection," in *Proc. ECCV*, 2016, pp. 450–466.
[45] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. CVPR*, 2015, pp. 5455–5463.
[46] J. Kim and V. Pavlovic, "A shape-based approach for salient object detection using deep learning," in *Proc. ECCV*, 2016, pp. 455–470.
[47] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. CVPR*, 2015, pp. 1265–1274.
[48] J. Kuen, Z. Wang, and G. Wang, "Recurrent attentional networks for saliency detection," in *Proc. CVPR*, 2016, pp. 3668–3677.
[49] N. Liu and J. Han, "Dhsnet: Deep hierarchical saliency network for salient object detection," in *Proc. CVPR*, 2016, pp. 678–686.
[50] R. Achanta, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels," EPFL, Lausanne, Switzerland, Tech. Rep. 149300, 2010.
[51] S. Yu, "Angular embedding: A robust quadratic criterion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 158–173, Jan. 2012.
[52] M. Maire, T. Narihira, and S. X. Yu, "Affinity CNN: Learning pixel-centric pairwise relations for figure/ground embedding," in *Proc. CVPR*, 2016, pp. 174–182.
[53] M. Maire, "Simultaneous segmentation and figure/ground organization using angular embedding," in *Proc. ECCV*, 2010, pp. 450–464.
[54] J. Kim, D. Han, Y.-W. Tai, and J. Kim, "Salient region detection via high-dimensional color transform," in *Proc. CVPR*, 2014, pp. 883–890.
[55] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. CVPR*, 2013, pp. 1155–1162.
[56] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. CVPR*, 2014, pp. 2814–2821.
[57] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Proc. CVPR*, 2014, pp. 280–287.
[58] V. Movahedi and J. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *Proc. CVPRW*, 2010, pp. 49–56.
[59] T. Liu *et al.*, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, Feb. 2011.
[60] S. Alpert, M. Galun, R. Basri, and A. Brandt, "Image segmentation by probabilistic bottom-up aggregation and cue integration," in *Proc. CVPR*, 2007, pp. 1–8.
[61] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. CVPR*, 2009, pp. 1597–1604.

Lihe Zhang received the M.S. degree in signal and information processing from Harbin Engineering University, Harbin, China, in 2001, and the Ph. D. degree in signal and information processing from the Beijing University of Posts and Telecommunications, Beijing, China, in 2004. He is currently an Associate Professor with the School of Information and Communication Engineering, Dalian University of Technology. His current research interests include computer vision and pattern recognition.

Jianwu Ai received the B.E. degree in electronic information engineering from the Harbin Engineering University, Harbin, China, in 2015. He is currently pursuing the M.S. degree with the School of Information and Communication Engineering, Dalian University of Technology, Dalian, China. His research interests include saliency detection.



Bowen Jiang received the M.S. degree in signal and information processing from the Dalian University of Technology, Dalian, China, in 2014. He is currently with AutoNavi Software Company, Ltd., Beijing, China. His current research interests include computer vision and machine learning with focus on object detection, image segmentation, and deep learning.

Huchuan Lu (SM'12) received the M.S. degree in signal and information processing and the Ph.D. degree in system engineering from the Dalian University of Technology (DUT), Dalian, China, in 1998 and 2008, respectively. He joined the faculty of the School of Information and Communication Engineering, DUT, in 1998, where he is currently a Full Professor. His current research interests include computer vision and pattern recognition with focus on visual tracking, saliency detection, and segmentation. He is a member of the ACM and an Associate Editor of the IEEE TRANSACTIONS ON CYBERNETICS.

Xiukui Li received the M.E. degree from the Department of Electrical Engineering, Beijing University of Posts and Telecommunications, Beijing, China, in 2004, and the Ph.D. degree from the Department of Electrical Engineering, Michigan Technological University, Houghton, MI, USA, in 2009.